REMARKS

Claims 1-26 are pending and under consideration. Claims 1-26 were rejected in the Office Action mailed May 4, 2006. In this Response, no claims are amended, added, or canceled.

I. Double Patenting Rejection of Claims 1-26

The Examiner has reinstated the Double Patenting rejection of claims 1-26 despite Applicants' filing of a terminal disclaimer with respect to co-pending application no. 10/005,783. The Examiner states that the attorney that signed the terminal disclaimer is not an attorney of record. That attorney, which is the undersigned, is associated with customer number 58328, to which the present application is also associated. Applicants therefore submit that the terminal disclaimer was signed by an attorney of record, and that the terminal disclaimer previously submitted is sufficient to require withdrawal of the double patenting rejection.

II. Obviousness Rejections under 35 U.S.C. §103(a)

Claims 1-4, 6-15, and 17-25 are rejected under 35 U.S.C. §103(a) as being allegedly unpatentable over *Calder et al.* (U.S. Patent No. 5,963,972, hereinafter "*Calder*") in view of *Lomet* (U.S. Patent No. 5,963,972), and further in view of *Razdow et al.* (U.S. Patent No. 6,330,008, hereinafter "*Razdow*"). Applicants respectfully traverse the rejection, and respectfully submit that he combination of *Calder* and *Lomet* fails to teach or suggest all of the limitations of independent claims 1, 8, 12, 19, 20, 23, and 24.

Regarding claim 1, the combination of *Calder*, *Lomet,* and *Razdow* fails to teach or suggest, for example, facilitating development of the data flow program by generating a graph representing the blocks and the determined dependencies and displaying the graph to a user. The

Examiner asserts that *Razdow* teaches generating and displaying such a graph, and refers to column 2, lines 38-44 and Figures 3 and 5. Applicants respectfully submit that the assertion is erroneous. The antecedent basis of the above limitation cannot be ignored. The term "blocks" refers to the blocks of memory, and the term "determined dependencies" refers to the dependencies among those blocks of memory. In contrast, the graph of *Razdow* includes objects that represent operators and lines that represent datalinks connecting the operators. See column 8, lines 45-56 and Figure 3 of *Razdow*. Thus, because the operators do not represent memory blocks and the datalinks do not represent dependencies among memory blocks, *Razdow* cannot teach or suggest this limitation. The combination of *Calder*, *Lomet,* and *Razdow* simply just does not suggest the generation and display of a graph depicting dependencies among memory regions.

Moreover, there is no motivation to combine *Calder*, *Lomet,* and *Razdow* as suggested by the Examiner. The Examiner asserts that the modification would be obvious because one of ordinary skill in the art would be motivated to minimize cache misses by ensuring the proper order of computer operations. However, this does not explain what benefit *Razdow* provides in this hypothetical combination. The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). Because *Calder* is directed to a code reordering algorithm for optimizing a cache and does not require human input or intervention, there is no reason for displaying *Calder*'s graph to a user. Thus, modifying *Calder* with *Razdow* offers no benefit, and therefore the desirability of the combination is not established.

For at least these reasons, *prima facie* obviousness has not been established, and claim 1 is patentable over the combination of *Calder*, *Lomet,* and *Razdow*. Claims 8, 12, 20, 23, and 24 also recite similar limitations, and are therefore patentable for at least the same reasons.

Regarding claim 2, the combination of *Calder*, *Lomet,* and *Razdow* fails to teach or suggest a graph comprising nodes assigned to the blocks and dependency arcs representing the determined dependencies. Unlike Applicants' claimed invention in which graph nodes are assigned to blocks of memory, *Calder's* graph nodes merely correspond to units of instructions -- the units of instructions do not correspond to blocks of memory. In response, the Examiner asserts that "a block is a group of instructions in a program that are treated as a unit. Therefore, the unit is store in one block of memory." The Examiner's assertion is incorrect. A block, in claim 1, is assigned both a portion of data and at least one code segment. Thus, *Calder's* assignment of nodes to units of instruction cannot be construed as an assignment of nodes to blocks of memory. Further, it is irrelevant that the unit of code instruction in *Calder* is stored in one block of memory, as alleged by the Examiner, because it is still the instruction that is assigned the node, not the block of memory. Accordingly, *Calder's* graph nodes do not correspond to blocks of memory and thus cannot teach or suggest this limitation.

For at least these reasons, *prima facie* obviousness has not been established, and claim 2 is patentable over the combination of *Calder*, *Lomet,* and *Razdow*. Claim 9 recites similar limitations and is therefore patentable for at least the same reasons.

Regarding claims 3, 4, 6, and 7, the Examiner contends that these limitations are taught by *Razdow* at column 3, lines 5-25 and column 4, lines 4-20, which are replicated below:

> In the preferred embodiment of the invention information about the flow of data between operator instances can be represented visual attributes of the lines representing datalinks. This further compacts the representation

> of complex parallel graphs, since it allows both datalinks and information about their performance to be represented in one line. The preferred embodiment can represent the data rate of an individual datalink by both the density and velocity of broken line segments along its line. This use of broken, moving line segments makes it easier to distinguish separate datalinks which are very close to each other or overlap in a visualization, since it allows the user to see through the broken parts of such lines. The ability to distinguish between such closely placed datalink lines is further increased by the fact that the velocity and segmentation pattern of such datalink lines varies.
>
> In the preferred embodiment of the invention, the performance monitor is provided with a data structure which identifies which processes executing on which processor nodes correspond to which instances of which operators in the parallel data flow graph being executed. The individual parallel processes involved in the execution send messages to the performance monitor which provide it with the information necessary to perform its visualization. The performance monitor is capable of sending data to individual parallel processors as they are executing the graph to vary the performance information they send. For example, the performance monitor can send one or more operators a message informing it to report information on the value of a particular field in the records supplied to it. This information can then be displayed in the monitor's visualization window.

Applicants respectfully submit that these passages clearly fail to teach or suggest any of presenting a dependency arc using an unsatisfied dependency visualization (claim 3), displaying in a visually distinctive manner unmet dependencies in a graph (claim 4), displaying unexecuted nodes using an unexecuted visualization (claim 6), and accentuating a portion of a data structure accessed by a code segment (claim 7).

With regard to claim 3, *Razdow* discusses visually representing datalinks between operators such that the performance of the datalinks may be visualized. These datalinks are not dependencies, the data rate over these datalinks cannot be construed as a "satisfied" or "unsatisfied" dependency. With regard to claim 4, *Razdow* discloses nothing that could be construed as "receiving a node selection specifying a selected one of the nodes." Furthermore,

as previously discussed, *Razdow* does not display dependencies among blocks of memory. With regard to claim 6, *Razdow* does not visually distinguish between executed and unexecuted nodes. With regard to claim 7, *Razdow* does not even disclose a data structure, let alone accentuating a portion of a data structure accessed by a code segment. If this rejection is maintained, Applicants respectfully request a more detailed explanation of the rejection wherein claim elements and limitations are mapped directly to the disclosure of *Razdow*.

Accordingly, *prima facie* obviousness has not been established, and claims 3, 4, 6, and 7 are patentable over *Calder*, *Lomet,* and *Razdow*. Claims 9-11, 14, 15, 17, 18, 21, and 22 recite similar limitations, and are therefore patentable for at least the same reasons.

Regarding claim 19, the combination of *Calder*, *Lomet,* and *Razdow* does not teach or suggest the generation and display of a graph depicting dependencies among memory regions, as previously discussed. Furthermore, the combination fails to teach or suggest "while the code segments are executing, determining which nodes in the graph are unexecuted nodes and which nodes in the graph are executed nodes; and displaying the unexecuted nodes in a manner visually distinctive from the executed nodes." As previously discussed with regard to claim 6, *Razdow* does not visually distinguish between executed and unexecuted nodes. Accordingly, *prima facie* obviousness has not been established, and claim 19 is patentable over *Calder*, *Lomet,* and *Razdow*.

Claims 5, 16, and 26 are rejected under 35 U.S.C. §103(a) as being allegedly unpatentable over *Calder* in view of *Lomet* and *Razdow*, and further in view of *Cai (U.S. Patent No. 6,349,363)*. Applicants respectfully traverse the rejection.

Claims 1, 12, and 24 are allowable as discussed above. *Cai* still fails to disclose or suggest Applicants' claimed data read and data write identifiers and fails to disclose or suggest

determining dependencies based on the data read and data write identifiers. Therefore, *Calder* in view of *Lomet* and *Cai* still fails to disclose or suggest claims 1, 12, and 24.

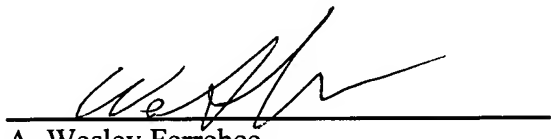
Claims 5, 16, and 26 depend directly or indirectly from claims 1, 12, or 24 and are therefore allowable for at least the same reasons that claims 1, 12, and 24 are allowable.

CONCLUSION

In view of the foregoing, it is submitted that claims 1-26 are patentable. It is therefore submitted that the application is in condition for allowance. Notice to that effect is respectfully requested.

Respectfully submitted,

Date: August 4, 2006



A. Wesley Ferrebee
Reg. No. 51,312

Customer No. 58328
SONNENSCHEIN NATH & ROSENTHAL LLP
P. O. Box 061080
Wacker Drive Station - Sears Tower
Chicago, Illinois 60606-1080
Telephone (202) 408-6832
Facsimile: (312) 876-7934